

## CLAIMS

I claim:

1. An image processing system comprising:

a color-space converter that is configured to provide a conversion of pixel values in a first color-space to corresponding pixel values in a second color-space,

a scaler that is configured to provide a scaling of pixel values at a first scale to corresponding pixel values at a second scale, and

a filter that is configured to apply a filter function to pixel values;

wherein

the color-space converter uses the filter to provide the conversion, and

the scaler uses the filter to provide the scaling.

2. The image processing system of claim 1, further including

a first multiplexer that is configured to selectively provide pixel values to the filter to selectively effect the conversion and the scaling.

3. The image processing system of claim 2, further including

a second multiplexer that is configured to selectively provide color-space-conversion coefficients and scaling coefficients to the filter to selectively effect the conversion and the scaling.

4. The image processing system of claim 3, wherein

a third multiplexer that is configured to selectively provide offset parameters to the filter to selectively effect the conversion and the scaling.

5. The image processing system of claim 1, wherein

the filter is an FIR filter.

6. The image processing system of claim 1, further including  
a memory that facilitates communication of pixel values among the color-space  
converter, the scaler, and the filter.

5 7. The image processing system of claim 1, wherein  
the filter is a 6-tap, 3-element FIR filter.

8. The image processing system of claim 1, wherein  
the filter includes a multiply-add array, and  
10 the color-space converter uses the multiply-add array of the filter to provide the  
conversion, and  
the scaler uses the multiply-add array of the filter to provide the scaling.

11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100

9. A processing system comprising:

a multiply-add array that includes:

six data inputs, each data input including three element inputs, and  
eighteen coefficient inputs;

5 the multiply-add array being configured to effect a 3x6 matrix multiplication of  
the six data inputs and the eighteen coefficients to provide an output having three element  
outputs, and

a multiplexer, operably coupled to the multiply-add array that is configured to provide,  
when the multiplexer is in a first selectable mode,

10 each of six pixel values, each having three component values, to each of  
the six data inputs; and

when the multiplexer is in a second selectable mode,

each of three components of a single pixel value to three of the six data  
inputs, each of the three components being provided to the three element inputs of each of the  
three data inputs, and

each of three offsets associated with the three components of the single  
pixel value to another three of the six data inputs, each of the three offsets being provided to the  
three element inputs of each of the another three data inputs.

10. The processing system of claim 9, wherein

the multiplexer is further configured to provide,

when the multiplexer is in the first selectable mode,

eighteen scaling coefficients to the eighteen coefficient inputs, to provide  
a scaled pixel value corresponding to the six pixel values as the output; and

25 when the multiplexer is in the second selectable mode,

nine color-space conversion coefficients to each of:

a first set of nine coefficient inputs of the eighteen coefficient  
inputs, and

30 a second set of another nine coefficient inputs of the eighteen  
coefficient inputs, to provide a color-space conversion of the single pixel value as the output.

